一，调节分析与中介分析

注意一下期中之前各种非参检验的关系，哪些是配对的，哪些是独立的

```
data$zscore = scale(data$adaptation)
data$group = 0
data$group[data$zscore > 1] = 1
data$group[data$zscore <= -1] = -1
data$group = factor(data$group, labels = c('low','medium','high'))
```

```
PROCESS(moderation_data, y="Loyalty", x="Relationship", mods="age_con")
PROCESS(mediation_data, y="Satisfaction", x="Relationship",
        meds="Discount",ci="boot", nsim=5000, seed=1)
```

X 和 M 均为连续变量；若 X 和 M 均为分类变量，则通过two-way anova分析处理 X*M 交互作用；若 X 为分类变量、M 为连续变量（或反之），则以 M 为协变量进行协方差分析

二，相关与距离

```
data1 <- data.frame(ms,age,sr)
ppcor::pcor(data1)
ppcor::spcor(data1)
```

欧式距离，切比雪夫距离，马氏距离

偏相关大于半偏相关/part

```
t=as.matrix(studiedData[,-5])
#euclidian distance
dist(t)  #unscaled
dist(scale(t)) #scaled, z scored
#cosine similarity
library(lsa)
cosine(t(t))
```

三，因子分析

```
# 对数据进行标准化处理（若题项量表一致则非必需）
IFDData_z <- as.data.frame(scale(IFDData))
# 检查数据中是否存在缺失值（必要时删除含缺失值的观测）
anyNA(IFDData_z)
IFDData_z <- IFDData_z[complete.cases(IFDData_z), ]
# 步骤2：数据检验
# 计算相关矩阵
Corr(IFDData_z)
# 执行巴特利特球形检验
cortest.bartlett(cor(IFDData_z), nrow(IFDData_z))
# 计算KMO检验值（采样充足度指标）
KMO(cor(IFDData_z))
# 提取相关系数矩阵
cor_matrix <- cor(IFDData_z)
# 计算相关矩阵的行列式（评估矩阵奇异性）
det(cor_matrix)
# 步骤3：确定提取的因子数量
# 绘制碎石图
scree(IFDData_z)
# 执行平行分析（比较实际数据与随机重采样数据的因子特征值）
fa.parallel(IFDData_z, fm = "pa", fa = "both")
# （注：需明确说明因子数量决策依据，此处以8个因子为例）
# 步骤4：执行因子提取
# 使用未加权最小二乘法（uls）提取8个因子，并进行方差最大旋转
fa_exact <- fa(IFDData_z, nfactors = 8, fm = 'uls',
               rotate = 'varimax')
# 绘制因子载荷图（可视化因子结构）
fa.diagram(fa_exact)
# 输出因子载荷矩阵（设置载荷阈值为0.4以辅助决策）
print(fa_exact$loadings, cutoff = 0.4)
# 分析完成！
```

四，信度与效度

Cronbach's α Inteternal

$$\alpha = \frac{k}{k-1}\left(1 - \frac{\sum_{i=1}^{k} S_i^2}{S_x^2}\right)$$

Guttman formula

$$\alpha = \frac{k}{k-1}\left(1 - \frac{\sum_{i=1}^{k} p_i q_i}{S_x^2}\right)$$

Split-half

$$r_{xx} = \frac{2r_{hh}}{1+r_{hh}}$$

$$r = 2\left(1 - \frac{s_a^2 + s_b^2}{s_x^2}\right)$$

经典测验理论（CTT）下的信度

- **公式**：$X = T + e$，其中X表示观测值，T表示真值，e表示测量误差。

  信度系数（reliability coefficient）$r_{xx} = \frac{S_T^2}{S_X^2}$，是真值方差与观测值方差的比值，通常真值方差（$T$）小于观测值方差（观测值方差受测量噪声影响）。

```
data10_z=as.data.frame(scale(data10))
RA_results_reversed <- jmv::reliability(data10_z, vars=
                        meanScale=TRUE, sdScale=TRUE, corPlot=TRUE,
alphaItems=TRUE,
                        meanItems=TRUE, sdItems=TRUE,
itemRestCor=TRUE,revItems=c('q02','q09','q19','q22','q23'))
RA_results$scale  # Reliability of the entire scale
RA_results$items  # cronbach's apha if this item is deleted;
RA_results$corPlot  # the inter-item correlation matrix
```

```
data10_x=data10_0[,c(-2,-9,-19,-22,-23,-3,-7,-12,-14)]
data10_z_x=as.data.frame(scale(data10_x))
new_df <- melt(data10_z_x,
               measure.vars = c('q01','q04', 'q05', 'q06', 'q08', 'q10',
'q11', 'q13','q15', 'q16', 'q17', 'q18', 'q20', 'q21'),
               variable.name = "item_factor",
               value.name = "y")
new_df$people_factor <- as.factor(rep(1:nrow(data10_z_x), ncol(data10_z_x)))
new_df$item_factor <- as.factor(new df$item factor)
lm0 = lm(y ~ ., data = new_df)
df.lm0 = df.residual(lm0)  # Residual degrees-of-freedom
MSE.lm0 = deviance(lm0)/df.lm0  # Mean sum of square
library(agricolae)  # need the nonadditivity test
res =  with(new_df, nonadditivity(y, people_factor, item_factor, df.lm0,
MSE.lm0))
```

```
custom_sorted_factor <- factor(factor_var, levels = c("C", "A", "B"))
```

```
data10_z_x <- as.data.frame(lapply(data10_z_x, function(x) {
attributes(x) <- NULL
return(x)}))
ICC_results = ICC(data10_z_x)
ICC_results$results
```

df$x39=6-df$x39

$$Y_{ij} = \mu + \alpha_i + \epsilon_{ij},$$

$$\text{ICC} = \frac{\sigma_\alpha^2}{\sigma_\alpha^2 + \sigma_\epsilon^2}.$$

五，聚类分析

```
plot(n_clusters(data11[,-1]))#确认分几类
data11_z=scale(data11[,-1])
data11_z = `row.names<-`(data11_z,data11$指标)
hclust(dist(data11_z)) %>%
  fviz_dend(k=3,rect = T,cex=0.8,horiz = T)
```

```
fviz_nbclust(df2, kmeans, method = "wss")
fviz_nbclust(df2, kmeans, method = "silhouette")
```

```
# Euclidean distance matrix
distMat = dist(df2, method = "euclidean")

#clustering, using the default method
hc = hclust(distMat)

# display dendrogram
plot(hc,labels=TRUE,hang=-1)
```

```
data2=data11[,-1]
kdf=data.frame(
  edu=apply(data2[c(9,7,5,8,6),],2,mean),
  tran=apply(data2[c(12,10,13,11,4,3,14),],2,mean),
  med=apply(data2[c(16,15,1,2),],2,mean))
resk = kmeans(kdf,centers = 2,iter.max = 20)
resk
fviz_cluster(resk, data = kdf)
```

initial_centers = temp[c('x1','x2','x3')] # use specified centers

六，生存分析

产生原因

未经历事件：研究期间部分个体始终未出现研究关注的事件，如在癌症复发研究中，部分患者直到研究结束都未出现癌症复发。

中途退出：个体从研究中主动退出，可能因个人意愿、不良反应等，退出后无法继续观察其事件发生情况。

其他原因死亡：个体在研究结束前死亡，但死亡原因并非研究关注的事件，例如研究某种疾病治疗效果时，患者因其他疾病死亡。

失访：在研究期间个体因各种原因失去联系，导致无法获取后续是否发生事件的信息。

类型

固定类型 I 删失：研究设定在特定时长 C 年后结束，研究期间未发生事件的个体，均在 C 年时被删失。比如一项为期 5 年的心血管疾病研究，5 年后未发生心血管事件的个体数据视为在第 5 年删失。

类型 II 删失：研究在达到预先设定的事件发生数量时结束，未发生事件的个体即被删失。例如计划观察到 100 例疾病复发事件后结束研究，此时剩余未复发个体数据被删失。

类型 III 删失（随机删失）：研究结束时，由于个体进入研究的时间不同，未发生事件个体的删失时间也各不相同。像一项长期随访研究，不同患者入组时间有先后，研究结束时，未发生事件患者的删失时间取决于各自入组时长 。

```
library(survival)
data12=read_xlsx("D:/HuaweiMoveData/Users/50376/Desktop/心统作业/death.xlsx")
data121=data12[data12$type==1,]
data122=data12[data12$type==2,]
surobj=Surv(event=data12$delta,time=data12$time)
surobj2 = Surv(event=data122$delta,time=data122$time)
surobj1 = Surv(event=data121$delta,time=data121$time)
median(survfit(surobj1~type,data121)$time)
surobj2 = Surv(event=data122$delta,time=data122$time)
median(survfit(surobj2~type,data122)$time)
survdiff(surobj~type,data12)
```

```
# survival analysis by Kaplan-Meier
# 18-02 肾上腺样瘤数据
kidney <- read_xlsx("data18-02.xlsx")
# R : The status indicator, normally 0=alive, 1=dead. Other choices are
TRUE/FALSE (TRUE = death) or 1/2 (2=death).
kidney$sta[kidney$sta == 9] <- 0
kidney$sta <- as.numeric(kidney$sta)
#计算生存曲线
survobj <- with(kidney, Surv(time,sta)) # 创建对象
fit0 <- survfit(survobj~tre+strata(k), data=kidney) # 构建生存分析模型
summary(fit0) # 查看结果，survival table
fit0  # check the median and 95% CL
surv_diff <- survdiff(survobj ~ tre,data = kidney,subset = c(kidney$k == 0)) #
logrank, test between survival curves, for k=0 condition
surv_diff
surv_diff$chisq
surv_diff <- survdiff(survobj ~ tre,data = kidney,subset = c(kidney$k == 1))
surv_diff
surv_diff$chisq
#画图
ggsurvplot(fit0, conf.int = T, surv.median.line = "hv")
ggsurvplot(fit0, conf.int = T, surv.median.line = "hv", fun = 'event')
```

```
# survival analysis by cox regression
lung2 <- read_xlsx("data18-03.xlsx")
# R : The status indicator, normally 0=alive, 1=dead. Other choices are
TRUE/FALSE (TRUE = death) or 1/2 (2=death).
lung2$status[lung2$status == 0] <- 2
lung2$status <- as.numeric(lung2$status)
#Cox regression
survobj <- with(lung2, Surv(time,status))
res.cox <- coxph(survobj ~ as.factor(therapy) + as.factor(cell) + kps +
diagtime + age + as.factor(prior), data = lung2, method = 'breslow', model =
TRUE, x = TRUE) #此处SPSS中应用的method为breslow，但R中默认为efron
summary(res.cox)  # beta, and overall test for the model (log likelihood ratio
test, logrank test)
ggsurvplot(survfit(res.cox), data = data)
```